

Block 2: Inference in BN

1. Introduction to inference in BN.
2. Exact inference.
3. Predicting a query variable.
4. Approximate inference with stochastic simulation.
 - ① Logic Sampling (LS).
 - ② Likelihood Weighting (LW).
 - ③ Assessing approximate inference algorithms (Kullback-Leibler divergence).
5. Soft evidence.
 - ① The “all things considered” method.
 - ② The “nothing else considered” method.
 - ③ Soft evidence as a noisy sensor.

1. Introduction to inference in BN

- Using a Bayesian network to compute (a posteriori) probability is called “*(Bayesian) inference*”.

When we enter evidence and use it to update the probabilities of the network we call it “*propagation*” or “*inference*” or “*belief updating*”.

- Inferences involve queries of the form (posterior probability):

$$P(X / E)$$



X = The query variable(s)

E = The evidence variable(s)

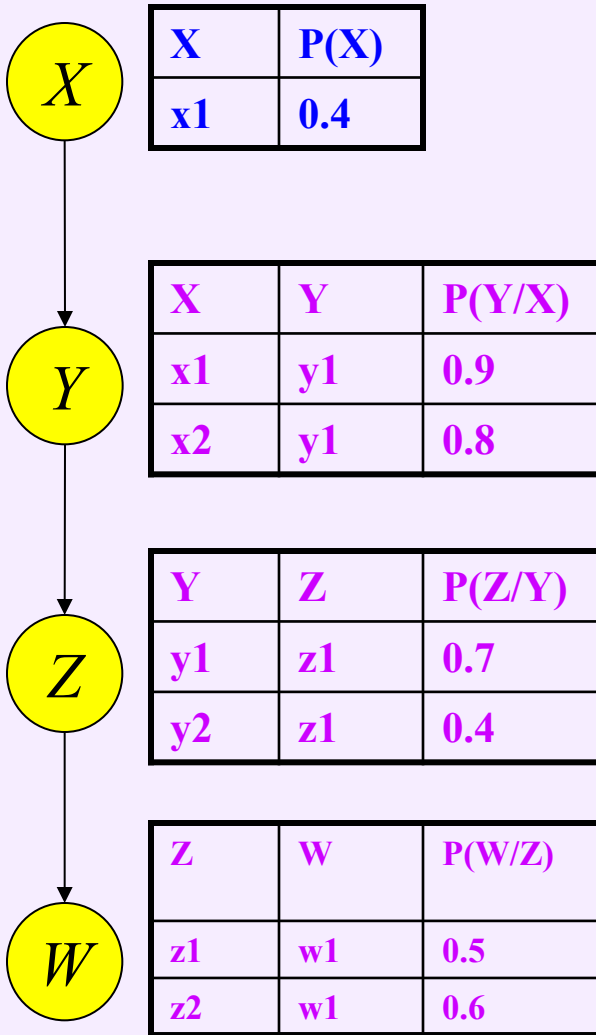
Variables of the BN that do not appear either as query variables or evidence variables are treated as **unobserved**.

- ✓ Exact inference is only feasible in small to medium-sized networks. In the introductory example (block 1, ELISA) we saw, as a standard application of Bayes' Theorem, how to perform inference in a two-node Bayesian network.
- ✓ Exact inference in large networks takes a very long time as larger BN address the problem of representing the joint probability distribution of a large number of variables.
- ✓ We resort to **approximate inference techniques** which are much faster and give pretty good results.

- **1980s:** researchers such as Lauritzen, Spiegelhalter and Pearl published algorithms that provided efficient propagation for a large class of BN models. These algorithms are very sophisticated and are efficient because they exploit the **BN structure** to carry out modular calculations rather than require calculations on the whole joint probability distribution. They are based on the idea of *message-passing* through the variables of the network. The most popular of them is the **junction tree algorithm**, that entails performing belief propagation on a modified graph called *junction tree*.
- The first commercial tool to implement an efficient propagation algorithm was developed in **1992** by **Hugin**, a Danish company.
- You need not concern yourselves with these algorithms since a number of packages for doing inference in BN have been developed: **Hugin**, **abn** or **bnlearn** of **R**, Netica, GeNIe, Elvira, BUGS,...

2. Exact Inference

Example 1

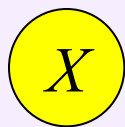


Consider this BN in which each r.v. takes only two values.

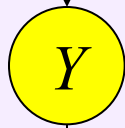
Conditional independencies entailed by the Markov condition:

Node	Parents	Nondesc.	Conditional Independency
X	Null set	Null set	None
Y	X	Null set	None
Z	Y	X	Z and X, given Y
W	Z	X, Y	W and {X, Y}, given Z

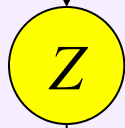
Then, first of all, the prior probabilities of all variables can be computed using the **Law of Total Probability** in the following way:



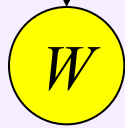
X	P(X)
x1	0.4



X	Y	P(Y/X)
x1	y1	0.9
x2	y1	0.8



Y	Z	P(Z/Y)
y1	z1	0.7
y2	z1	0.4



Z	W	P(W/Z)
z1	w1	0.5
z2	w1	0.6

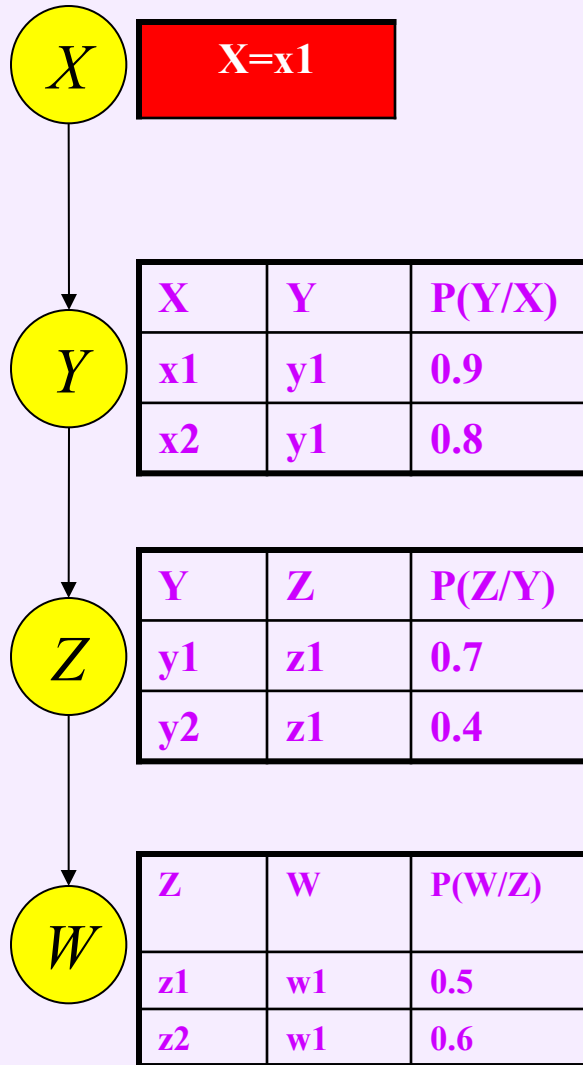
$$P(y_1) = P(y_1/x_1) P(x_1) + P(y_1/x_2) P(x_2) \\ = 0.9 \times 0.4 + 0.8 \times 0.6 = 0.84$$

$$P(z_1) = P(z_1/y_1) P(y_1) + P(z_1/y_2) P(y_2) \\ = 0.7 \times 0.84 + 0.4 \times 0.16 = 0.652$$

$$P(w_1) = P(w_1/z_1) P(z_1) + P(w_1/z_2) P(z_2) \\ = 0.5 \times 0.652 + 0.6 \times 0.348 = 0.5348$$

Note that the computation for each variable requires information determined for its parent.

We can consider this method a message-passing algorithm in which each node passes its child a message needed to compute the child's probabilities.

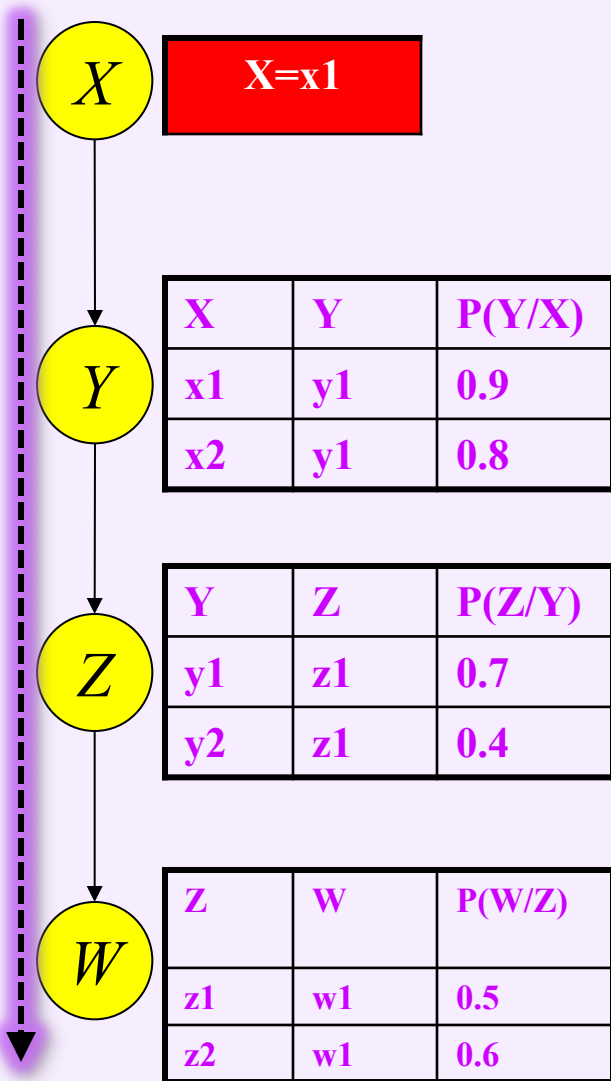


Suppose now that X is instantiated for x_1 . Since the Markov condition entails that each variable is conditionally independent of X given its parent, we can compute the conditional probabilities for the remaining variables by using the [Law of Total Probability](#).

$$P(y_1/x_1) = 0.9$$

$$\begin{aligned}
 P(z_1/x_1) &= P(z_1/y_1, x_1) P(y_1/x_1) + P(z_1/y_2, x_1) P(y_2/x_1) \\
 &= P(z_1/y_1) P(y_1/x_1) + P(z_1/y_2) P(y_2/x_1) \\
 &= 0.7 \times 0.9 + 0.4 \times 0.1 = 0.67
 \end{aligned}$$

$$\begin{aligned}
 P(w_1/x_1) &= P(w_1/z_1, x_1) P(z_1/x_1) + P(w_1/z_2, x_1) P(z_2/x_1) \\
 &= P(w_1/z_1) P(z_1/x_1) + P(w_1/z_2) P(z_2/x_1) \\
 &= 0.5 \times 0.67 + 0.6 \times (1 - 0.67) = 0.533
 \end{aligned}$$

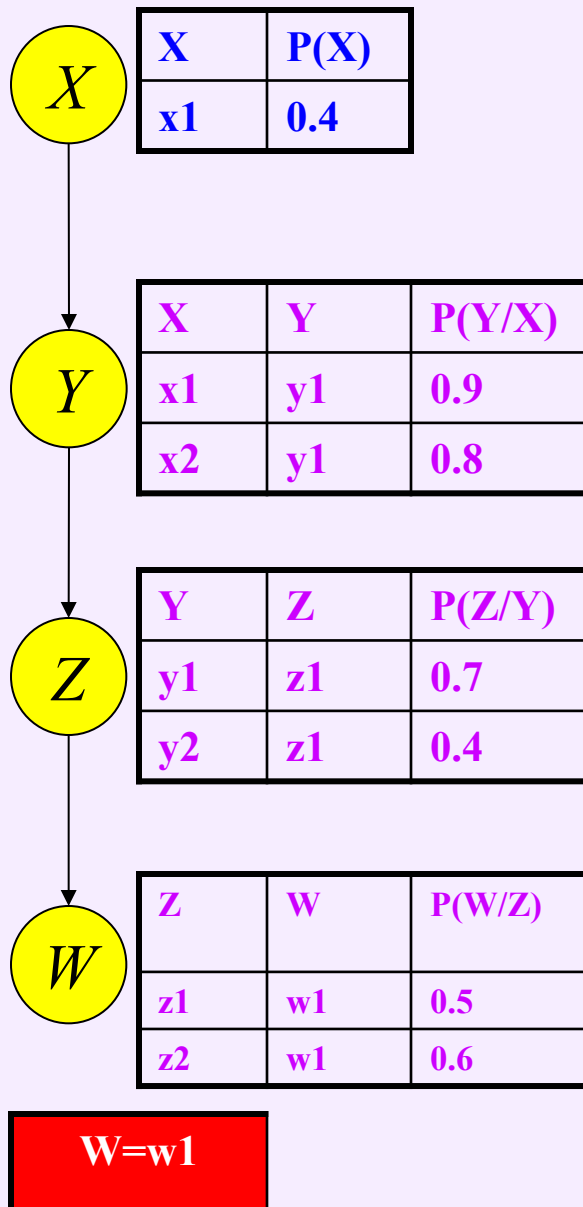


This example shows how we can use **downward propagation** of messages to compute the conditional probabilities of variables **below** the instantiated variable.

Instantiated: $X=x1$

Computed: $P(y1/x1)=0.9$
 $P(z1/x1)=0.67$
 $P(w1/x1)=0.533$

Can we compute conditional probabilities of variables **above** the instantiated variable?



Suppose now instead that W is instantiated for $w1$ (and no other variable is instantiated). We can use **upward propagation** of messages to compute the conditional probabilities of variables **above** the instantiated variable.

First, we use Bayes' Theorem to compute

$$\begin{aligned}
 P(z_1/w_1) &= \frac{P(w_1/z_1) P(z_1)}{P(w_1)} \\
 &= \frac{0.5 \times 0.652}{0.5348} = 0.6096
 \end{aligned}$$

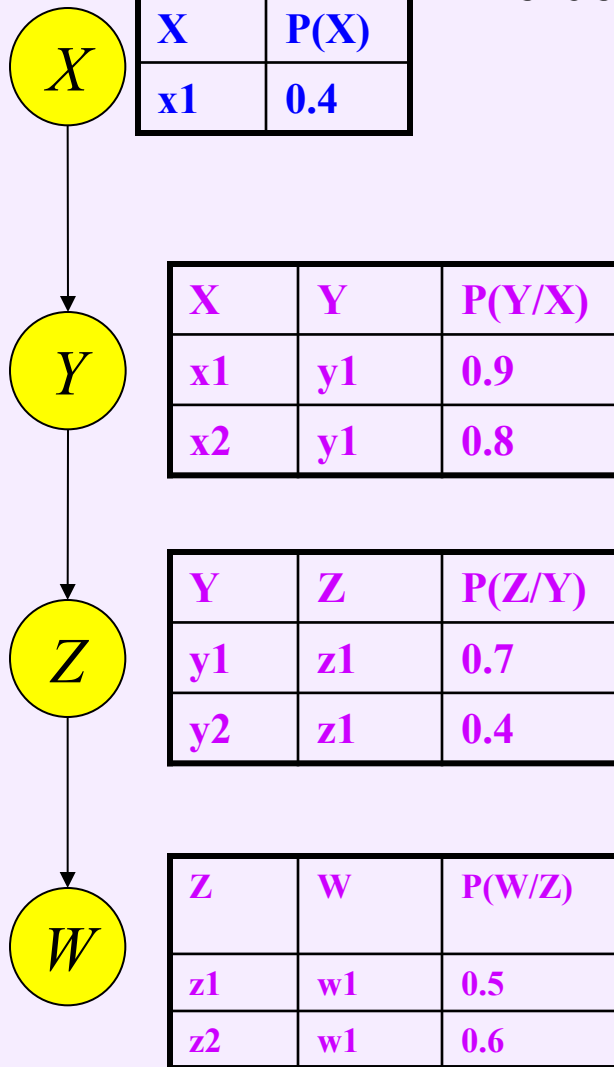
To compute $P(y_1/w_1)$, again apply Bayes' Theorem:

$$P(y_1/w_1) = \frac{P(w_1/y_1) P(y_1)}{P(w_1)}$$

But we cannot yet complete this computation because we do not know $P(w_1/y_1)$. We can obtain this value using downward propagation as follows:

$$\begin{aligned} P(w_1/y_1) &= P(w_1/z_1) P(z_1/y_1) + P(w_1/z_2) P(z_2/y_1) \\ &= 0.5 \times 0.7 + 0.6 \times 0.3 = 0.53 \end{aligned}$$

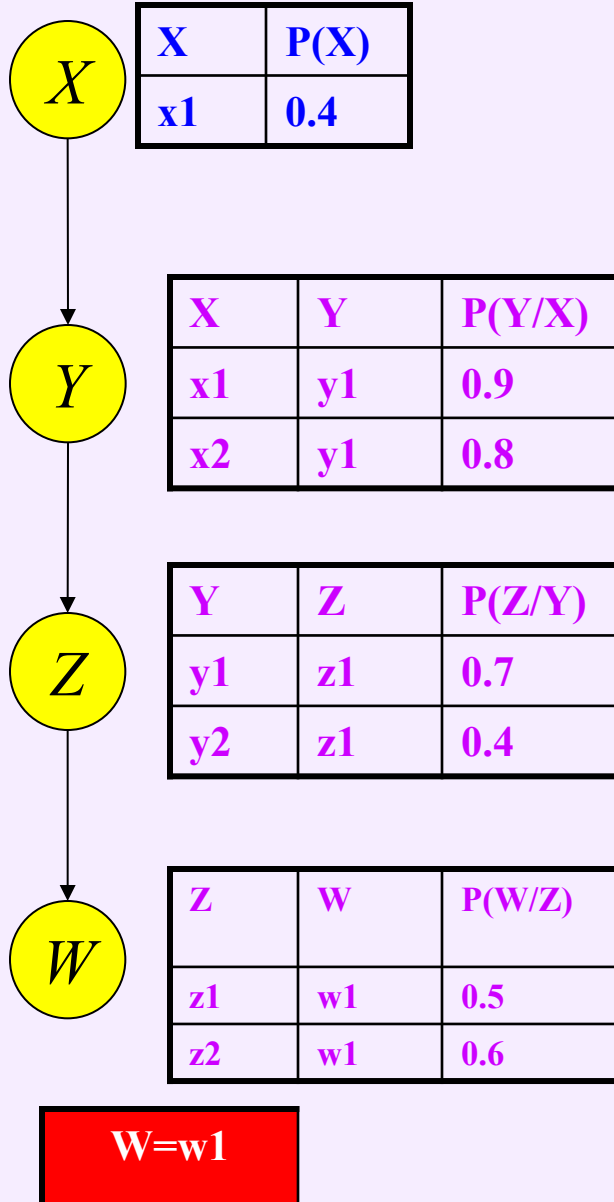
$$\begin{aligned} P(w_1/y_2) &= P(w_1/z_1) P(z_1/y_2) + P(w_1/z_2) P(z_2/y_2) \\ &= 0.5 \times 0.4 + 0.6 \times 0.6 = 0.56 \end{aligned}$$



W=w1

(we also compute $P(w_1/y_2)$ because X will need this value)

Then, by using that $P(w_1/y_1)=0.53$,



$$P(y_1/w_1) = \frac{P(w_1/y_1) P(y_1)}{P(w_1)}$$

$$= \frac{0.53 \times 0.84}{0.5348} \approx 0.83246$$

Finally, we compute

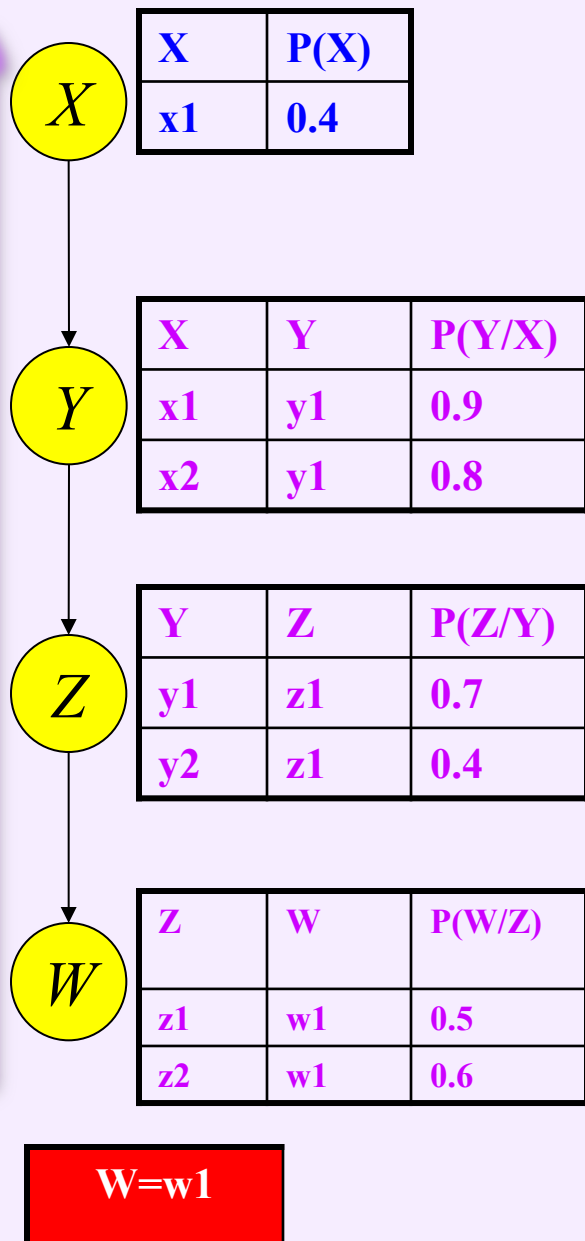
$$P(x_1/w_1) = \frac{P(w_1/x_1) P(x_1)}{P(w_1)}$$

$$P(w_1/x_1) = P(w_1/y_1) P(y_1/x_1) + P(w_1/y_2) P(y_2/x_1)$$

$$= 0.53 \times 0.9 + 0.56 \times 0.1 = 0.533$$

$$P(x_1/w_1) = \frac{P(w_1/x_1) P(x_1)}{P(w_1)}$$

$$= \frac{0.533 \times 0.4}{0.5348} \approx 0.39865$$



This example shows how we can use **upward propagation** of messages to compute the conditional probabilities of variables **above** the instantiated variable.

Instantiated: $W=w1$

Computed: $P(z1/w1)=0.6096$

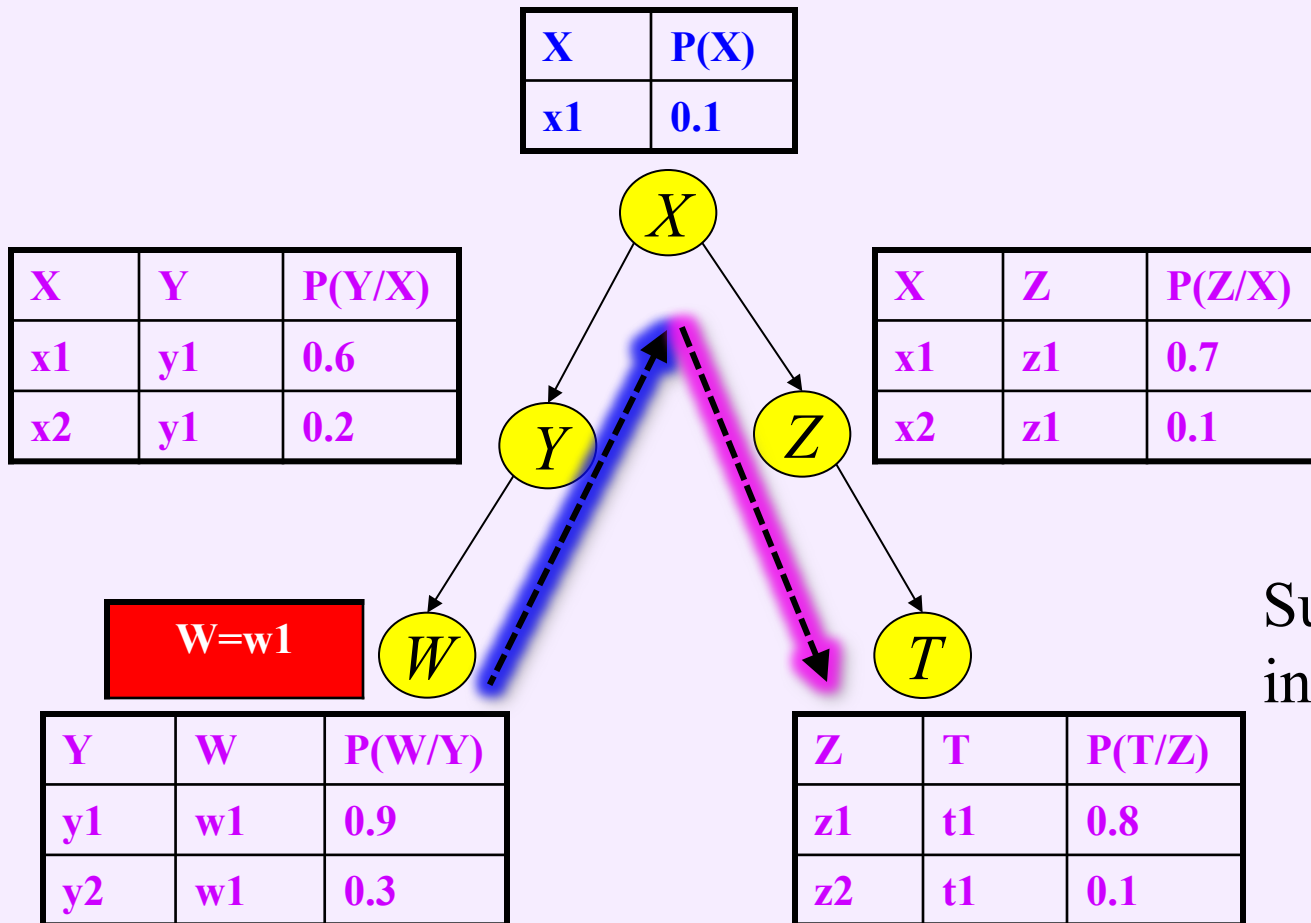
$P(y1/w1) \approx 0.83246$

$P(w1/x1) \approx 0.39865$

How can we turn corners in a tree?

Example 2

Consider this BN in which each r.v. has only two possible values.



Suppose W is instantiated for $w1$.

- ✓ Exercise: Compute $P(y1/w1)$ and $P(x1/w1)$ using the **upward propagation algorithm**, and then $P(z1/w1)$ and $P(t1/w1)$ using the **downward propagation algorithm**.

3. Predicting a query variable

Given known the value(s) of the evidence variable(s) E , we would like to give a prediction for a query variable X .

The prediction of X given E is the instantiation of X with the largest posterior probability. That is, if x_1, \dots, x_n are the possible instantiations of X (values that variable X takes), then

$$x^* = \arg \max_{i=1, \dots, n} P(X = x_i / E)$$

is the prediction for X and

$$P(X = x^* / E)$$

is said to be the *confidence level (CL) of the prediction*.

4. Approximate Inference with Stochastic Simulation

In general, exact inference in Bayesian Networks is computationally complex. For large networks (or networks densely connected), approximate algorithms must be used.

One approach to approximate inference for multiply-connected networks is **stochastic simulation**, which uses the network to generate a large number of cases from the network distribution, which in turn are used to estimate the posterior probability of a query node given the evidence. As more cases are generated, the estimate converges on the exact probability.

As with exact inference, there is a computational complexity. However, in practice, if the evidence being conditioned upon is not too unlikely, the approximate converges fairly quickly!

① Logic Sampling (LS)

The simplest algorithm is that of Logic Sampling (LS) (Henrion, 1988).

This algorithm generates a case by randomly selecting values for each node, weighted by the probability of that value occurring.

The nodes are traversed from the root nodes down to the leaves, so at each step the weighting probability is either the prior or the CPT entry for the sampled parent values.

When all the nodes have been visited, we have a “case” (i.e. an instantiation of all the nodes of the BN).

To estimate $P(X=xi / E=e)$ we compute the ratio of cases where both $X=xi$ and $E=e$ are true to the number of cases where just $E=e$ is true. So, after the generation of each case, these combinations are counted, as appropriate.

Main problem: if evidence $E=e$ is unlikely. Then, most of the cases have to be discarded, as they don't contribute to the run counts.

The Logic Sampling (LS) Algorithm

Initialization:

- Create a count variable $\text{Count}(xi, e)$ and a count variable $\text{Count}(e)$.
- Initialize both count variables to 0.

For each round of simulation:

1. For all root nodes:

Randomly choose a value for it, weighting the choice by the priors.

2. Loop:

Choose values randomly for children, using the conditional probabilities given the known values of the parents.

Until all the leaves are reached.

3. Update run counts:

If the case includes $E=e$, $\text{Count}(e) \leftarrow \text{Count}(e)+1$

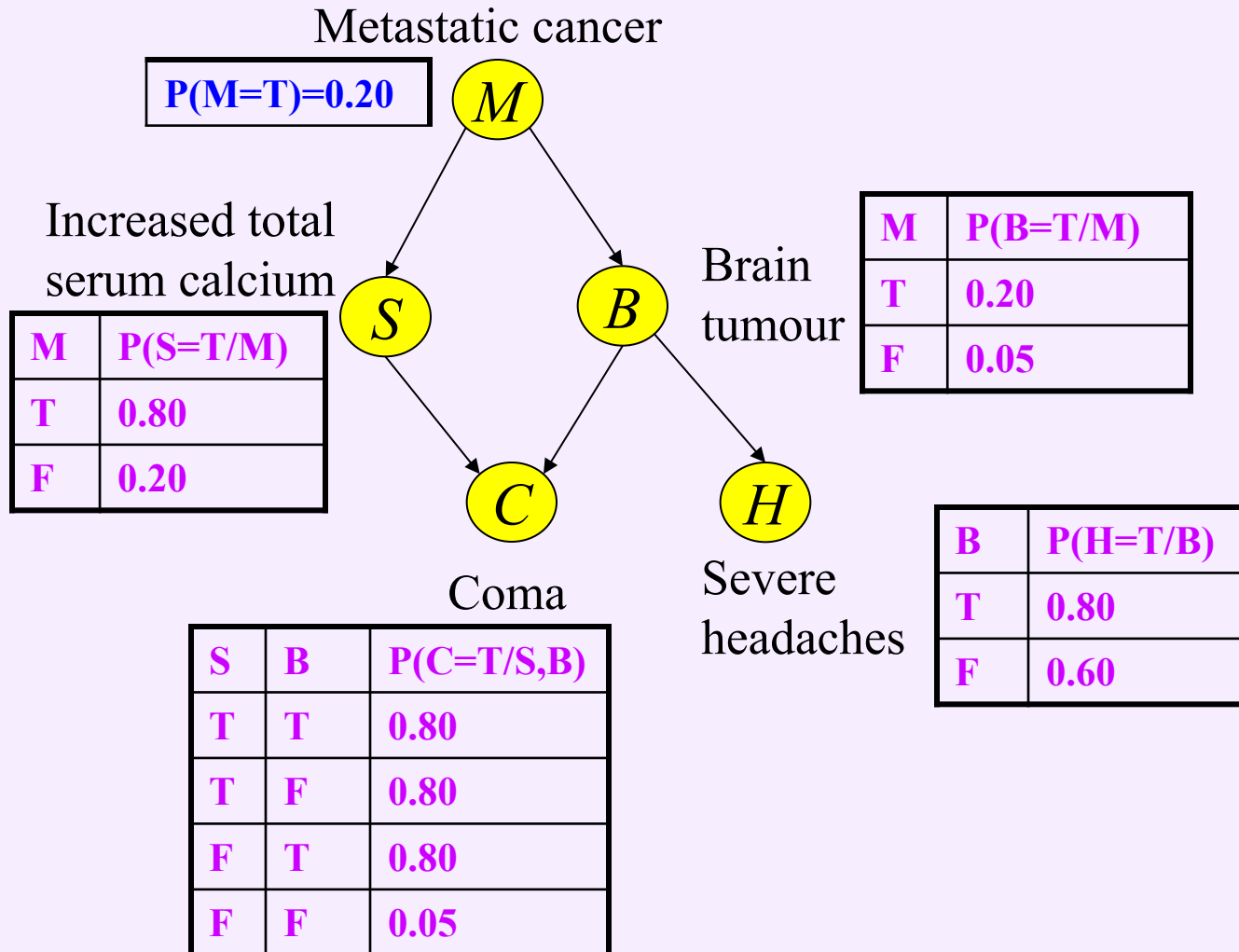
If the case includes $X=xi$ and $E=e$, $\text{Count}(xi, e) \leftarrow \text{Count}(xi, e)+1$

Current estimate for the posterior probability:

$$P(X=xi / E=e) = \text{Count}(xi, e) / \text{Count}(e)$$

Example 3: the Metastatic cancer network

The two causes of C (S and B) share a common parent M . Consider this BN in which each r.v. has only two possible values (T/F).



Let us work through one round of the Logic Sampling algorithm simulation for estimate $P(M=T / H=T)$ and/or $P(M=F / H=T)$.

- M is the only root node, which has prior $P(M=T)=0.2$. The random number generator produces a value between 0 and 1; any number > 0.2 means F is selected. Suppose that is the case.
- Next, the values for children S and B must be chosen, using the CPT entries $P(S=T / M=F)=0.20$ and $P(B=T / M=F)=0.05$. Suppose the values $S=T$ and $B=F$ are chosen randomly.
- Finally, the values for C and H must be chosen weighted with the probabilities $P(C=T / M=F, S=T, B=F) = P(C=T / S=T, B=F)=0.80$ and $P(H=T / M=F, S=T, B=F) = P(H=T / B=F)=0.60$. Suppose that values $C=F$ and $H=T$ are chosen randomly.
- Then, the full “case” for this simulation round is:
$$M=F, S=T, B=F, C=F \text{ and } H=T$$
- This case would add one to the count variable $\text{Count}(H=T)$, one to $\text{Count}(M=F, H=T)$, but **not** to $\text{Count}(M=T, H=T)$.

② Likelihood Weighting (LW)

A modification of the LS algorithm (Fung and Chang, 1989; Shachter and Peot, 1989).

This algorithm overcomes the problem with unlikely evidence in the LS algorithm, always employing the sampled value for each evidence node.

But instead of adding “1” to the run count, the CPTs for the evidence node (or nodes) are used to determine how likely that evidence is (or that evidence combination is) given the parent state, and that fractional **likelihood** is the number added to the run count.

Initialization: The Likelihood Weighting (LW) Algorithm

- Create a count variable $\text{Count}(x_i, e)$ and a count variable $\text{Count}(e)$
- Initialize both count variables to 0.

For each round of simulation:

1. For all root nodes:
 - If a root is the evidence node E ,
Choose $E=e$, and assign $\text{likelihood}(E=e) \leftarrow P(E=e)$
 - Else
Randomly choose a value for it, weighting the choice by the priors.
 2. Loop:
 - If a child is the evidence node E ,
Choose $E=e$, and $\text{likelihood}(E=e) \leftarrow P(E=e / \text{chosen parents values})$
 - Else
Choose values randomly for children, using the conditional probabilities given the known values of the parents.
- Until all the leaves are reached.

3. Update run counts:

If the case includes $E=e$,

$$\text{Count}(e) \leftarrow \text{Count}(e) + \text{likelihood}(E=e)$$

If the case includes $X=x_i$ and $E=e$,

$$\text{Count}(x_i, e) \leftarrow \text{Count}(x_i, e) + \text{likelihood}(E=e)$$

Current estimate for the posterior probability:

$$P(X=x_i / E=e) = \text{Count}(x_i, e) / \text{Count}(e)$$

Note: if the evidence $E=e$ is a “combination of evidences” consisting of $\{E_1=e_1, \dots, E_n=e_n\}$, steps 1. and 2. are the same, doing them for each $j=1, \dots, n$. The only difference is in step 3:

3. Update run counts:

If the case includes $E=e$,

$$\text{Count}(e) \leftarrow \text{Count}(e) + \prod_j \text{likelihood}(E_j=e_j)$$

If the case includes $X=x_i$ and $E=e$,

$$\text{Count}(x_i, e) \leftarrow \text{Count}(x_i, e) + \prod_j \text{likelihood}(E_j=e_j)$$

Example 3 (revisited)

Let us work through one round of the Likelihood Weighting algorithm simulation for estimate $\mathbf{P(C=T / B=T)}$.

- Choose a value for M (the only root node) with prior $P(M=T)=0.2$. Assume we choose $M=F$ randomly.
- Next, the values for child S must be chosen, using the CPT entry $P(S=T / M=F)=0.20$. Assume $S=T$ is chosen.
- B is an evidence node, which has been to been set to T , and $P(B=T / M=F)=0.05$. So this run counts as 0.05 of a complete run.
- Finally, the value for C must be chosen weighted with the probabilities $P(C=T / M=F, S=T, B=T) = P(C=T / S=T, B=T)=0.80$. Assume $C=T$.
(Note that in this case we don't need to choose a value for node H).
- We have completed a run with likelihood 0.05 that reports $C=T$ given $B=T$. Hence, both $\text{Count}(B=T)$ and $\text{Count}(C=T, B=T)$ are incremented in 0.05.

③ Assessing approximate inference algorithms (Kullback-Leibler divergence)

In order to assess the performance of a particular approximate inference algorithm, or to compare algorithms, we use a measure for the quality of the solution at any particular time: the Kullback-Leibler divergence, which can only be applied when the network is such that the exact posterior probability can be computed.

DEFINITION: The Kullback-Leibler divergence of the updated belief for a query node X given an evidence $E=e$ is:

$$KL(P(X/E = e), P'(X/E = e)) = \sum_{x_i} P(X = x_i/E = e) \log \frac{P(X = x_i/E = e)}{P'(X = x_i/E = e)}$$

where $P(X=x_i/E=e)$ are computed by an exact algorithm, and $P'(X=x_i/E=e)$ by the approximate algorithm.

PROPERTY: The Kullback-Leibler divergence is non-symmetric, non-negative (with the convention $0 \log(0)=0$ and $0 \log(0/0)=0$), could be $+\infty$ and is $= 0$ if and only if for all x_i ,

$$P(X = x_i / E = e) = P'(X = x_i / E = e)$$

Proof: Denote $g(x) = \frac{P(X = x / E = e)}{P'(X = x / E = e)} \in [0, +\infty]$ and $\varphi(t) = t \log(t)$. Then,

$$\begin{aligned} KL(P(X/E = e), P'(X/E = e)) &= \sum_{x_i} P(X = x_i / E = e) \log \frac{P(X = x_i / E = e)}{P'(X = x_i / E = e)} \\ &= \sum_{x_i} \varphi(g(x_i)) P'(X = x_i / E = e). \end{aligned} \quad (1)$$

By Taylor's development around 1, since $\varphi(1) = 0$,

$$\begin{aligned} \varphi(g(x)) &= \varphi(1) + (g(x) - 1) \varphi'(1) + \frac{1}{2} (g(x) - 1)^2 \varphi''(h(x)) \\ &= (g(x) - 1) \varphi'(1) + \frac{1}{2} (g(x) - 1)^2 \varphi''(h(x)) \end{aligned} \quad (2)$$

where $h(x)$ lies between $g(x)$ and 1 (then, $h(x) \in (0, +\infty)$).

Therefore, by (2),

$$\begin{aligned}
& \sum_{x_i} \varphi(g(x_i)) P'(X = x_i/E = e) = \\
& \sum_{x_i} \left((g(x_i) - 1) \varphi'(1) + \frac{1}{2} (g(x_i) - 1)^2 \varphi''(h(x_i)) \right) P'(X = x_i/E = e) = \\
& \sum_{x_i} (g(x_i) - 1) \varphi'(1) P'(X = x_i/E = e) + \\
& \sum_{x_i} \frac{1}{2} (g(x_i) - 1)^2 \varphi''(h(x_i)) P'(X = x_i/E = e) = \\
& \sum_{x_i} \frac{1}{2} (g(x_i) - 1)^2 \varphi''(h(x_i)) P'(X = x_i/E = e), \tag{3}
\end{aligned}$$

since

$$\begin{aligned}
& \sum_{x_i} (g(x_i) - 1) \varphi'(1) P'(X = x_i/E = e) = \\
& \varphi'(1) \sum_{x_i} g(x_i) P'(X = x_i/E = e) - \varphi'(1) \sum_{x_i} P'(X = x_i/E = e) = \\
& \varphi'(1) \times 1 - \varphi'(1) \times 1 = 0,
\end{aligned}$$

which is a consequence of the fact that

$$\sum_{x_i} g(x_i) P'(X = x_i/E = e) = \sum_{x_i} P(X = x_i/E = e) = \sum_{x_i} P'(X = x_i/E = e) = 1.$$

By replacing (3) into (1), and taking into account that $\varphi''(t) = 1/t$, which is > 0 because $t \in (0, +\infty)$, we have that

$$KL(P(X/E = e), P'(X/E = e)) = \sum_{x_i} \frac{1}{2} (g(x_i) - 1)^2 \varphi''(h(x_i)) P'(X = x_i/E = e) \geq 0 \quad \square$$

Often assessing the performance of a particular approximate inference algorithm, or comparing the performance of approximate inference algorithms, involves plotting the KL divergence against the number of iterations.

By this procedure it has been confirmed that:

1. LW algorithm has faster convergence compared with LS algorithm.
2. Stochastic simulation methods perform better when evidence is nearer to root nodes.

5. Soft evidence

Two types of evidence: **hard evidence** and **soft evidence**.

Hard evidence

Is information to the effect that some event has occurred, which is also the type of evidence we have considered previously.

Soft evidence

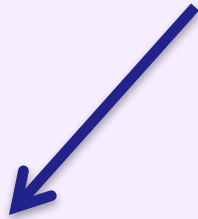
Soft evidence is not conclusive: we may get an unreliable testimony that event A occurred, which may increase our belief in A but not to the point where we would consider it certain.

For example, our neighbour who is known to have a hearing problem may call to tell us they heard the alarm trigger in our home. Such a call may not be used to categorically confirm the event Alarm but can still increase our belief in Alarm to some new level.

One of the key issues relating to soft evidence is
how to specify its strength.

There are two main methods:

The “all things considered” method



This method for specifying soft evidence on event A is by stating the new belief in A after the evidence has been accommodated.

The “nothing else considered” method



The second method for specifying soft evidence on event A is based on declaring the strength of this evidence independently of currently held beliefs.

① The “all things considered” method

This method for specifying soft evidence on event A is by stating the new belief in A after the evidence has been accommodated.

For example, we would say “after receiving my neighbour’s call, my belief in the alarm triggering stands now at 0.85”. Formally, we are specifying soft evidence as a constraint $P'(A)=q$ ($=0.85$), where P' denotes the new state of belief after accommodation the evidence, and A is the event to which the evidence pertains.

This is known as the “**all things considered**” method since the new belief in A depends not only on the strength of evidence but also on our initial beliefs that existed before the evidence was obtained.

That is: $P'(A)=q$ is not a statement about the strength of evidence per se but about the result of its integration with our initial beliefs.

Suppose we obtain some soft evidence on event A that leads us to change our belief in A to q . Then, $P'(A)=q$, $P'(\bar{A})=1-q$. Therefore:

$$P'(\omega) = \begin{cases} \frac{q}{P(A)} P(\omega) & \text{if } \omega \in A \\ \frac{1-q}{P(A^c)} P(\omega) & \text{if } \omega \in A^c \end{cases}$$

And moreover,

$$\begin{aligned} P'(B) &= \sum_{\omega \in B} P'(\omega) = \sum_{\omega \in B \cap A} P'(\omega) + \sum_{\omega \in B \cap A^c} P'(\omega) \\ &= \sum_{\omega \in B \cap A} \frac{q}{P(A)} P(\omega) + \sum_{\omega \in B \cap A^c} \frac{1-q}{P(A^c)} P(\omega) \\ &= q \frac{P(B \cap A)}{P(A)} + (1-q) \frac{P(B \cap A^c)}{P(A^c)} \\ &= q P(B/A) + (1-q) P(B/A^c) \end{aligned}$$

Where $P'(B)$ is the new state of belief on event B after accommodating the soft evidence $P'(A)=q$.

This method of updating a state of belief in the face of soft evidence is known as **Jeffrey's rule**:

$$P'(B) = q P(B/A) + (1 - q) P(B/A^c)$$

(Note that Bayes conditioning is a special case of Jeffrey's rule when $q=1$, in which case $P'(B)=P(B/A)$, that is, we have **hard evidence**)

Generalization of **Jeffrey's rule** to the case where the soft evidence concerns a set of mutually exclusive and exhaustive events A_1, \dots, A_n .

$$P'(B) = \sum_{i=1}^n q_i P(B/A_i)$$

Where $q_i=P'(A_i)$ is the soft evidence on event A_i , $i=1, \dots, n$.

Generalized Jeffrey's rule:

$$P'(B) = \sum_{i=1}^n q_i P(B/A_i)$$

Example: assume that we are given a piece of cloth C where its color can be one of green (cg), blue (cb), or violet (cv). We want to know whether the next day the cloth will be sold (s) or not sold (ns). Our original state of belief is as follows:

Sold	Color	P()
s	cg	0.12
ns	cg	0.18
s	cb	0.12
ns	cb	0.18
s	cv	0.32
ns	cv	0.08

Therefore, our original belief in the cloth being sold is

$P(s)=0.12+0.12+0.32=0.56$, and our original beliefs in the colors are $P(cg)=0.30$, $P(cb)=0.30$ and $P(cv)=0.40$.

Sold	Color	P()
s	cg	0.12
ns	cg	0.18
s	cb	0.12
ns	cb	0.18
s	cv	0.32
ns	cv	0.08

$P(s)=0.56$, $P(cg)=0.30$, $P(cb)=0.30$, $P(cv)=0.40$

Assume that we now inspect the cloth by candlelight and we conclude that our new beliefs in these colors should be

$P'(cg)=0.70$, $P'(cb)=0.25$ and $P'(cv)=0.05$

If we apply Jeffrey's rule we get for the event $A=s$:

$$\begin{aligned}
 P'(s) &= P'(cg) P(s/cg) + P'(cb) P(s/cb) + P'(cv) P(s/cv) \\
 &= 0.70 \times \frac{0.12}{0.12 + 0.18} + 0.25 \times \frac{0.12}{0.12 + 0.18} + 0.05 \times \frac{0.32}{0.32 + 0.08} \\
 &= 0.42
 \end{aligned}$$

$P(s)=0.56$  $P'(s)=0.42$

Sold	Color	P()
s	cg	0.12
ns	cg	0.18
s	cb	0.12
ns	cb	0.18
s	cv	0.32
ns	cv	0.08


$P(s)=0.56$, $P(cg)=0.30$, $P(cb)=0.30$, $P(cv)=0.40$

Soft evidence:

$P'(cg)=0.70$, $P'(cb)=0.25$ and $P'(cv)=0.05$

The full new state of belief according to

Jeffrey's rule is:



Sold	Color	P'()
s	cg	$0.70 \times 0.12 / 0.30 = 0.28$
ns	cg	$0.70 \times 0.18 / 0.30 = 0.42$
s	cb	$0.25 \times 0.12 / 0.30 = 0.10$
ns	cb	$0.25 \times 0.18 / 0.30 = 0.15$
s	cv	$0.05 \times 0.32 / 0.40 = 0.04$
ns	cv	$0.05 \times 0.08 / 0.40 = 0.01$

Note how the new belief is simply a scaled version of the old belief with 3 different scaling constants corresponding to cg, cb, cv.

$$\begin{aligned}
 P'(s, cg) &= P(s, cg/cg) P'(cg) + P(s, cg/cb) P'(cb) + P(s, cg/cv) P'(cv) \\
 &= P(s, cg/cg) P'(cg) + 0 + 0 \\
 &= P(s/cg) P'(cg) = \frac{0.12}{0.30} 0.70 = 0.28
 \end{aligned}$$

② The “nothing else considered” method

The second method for specifying soft evidence on event A is based on declaring the strength of this evidence independently of currently held beliefs.

Let us define the odds of the event A as follows:

$$\text{Odds}(A) = \frac{P(A)}{P(A^c)}$$

“Odds in favour of A ”

An odds of 1 indicates that we believe A and no A equally, while an odds of 10 indicates that we believe A ten times more than we believe no A .

Given the notion of odds, we can specify soft evidence on event A by declaring the relative change it induces on the odds of A , that is, the ratio:

$$k = \frac{Odds'(A)}{Odds(A)}$$

where

$$Odds'(A) = \frac{P'(A)}{P'(A^c)}$$

“Bayes factor”

- ✓ A Bayes factor of **1** indicates neutral soft evidence and a Bayes factor of **2** indicates soft evidence on A that is strong enough to double the odds of A.
- ✓ As the Bayes factor tends to **infinity**, the soft evidence tends toward hard evidence **confirming** A.
- ✓ As the factor tends to **zero**, the soft evidence tends toward hard evidence **refuting** A.

This method of specifying evidence is sometimes known as the “nothing else considered” method as it is a statement about the strength of evidence without any reference to the initial state of belief.

Suppose that we obtain soft evidence on A whose strength is given by a Bayes factor of k, and our goal is to compute the new state of belief P' that results from accommodating this evidence.

$$k = \frac{Odds'(A)}{Odds(A)} = \frac{P'(A)/P'(A^c)}{P(A)/P(A^c)}$$

$$\Rightarrow k \frac{P(A)}{P(A^c)} = \frac{P'(A)}{1 - P'(A)}$$

$$\Rightarrow k \frac{P(A)}{P(A^c)} - k \frac{P(A)}{P(A^c)} P'(A) = P'(A)$$

$$\Rightarrow P'(A) = \frac{k \frac{P(A)}{P(A^c)}}{1 + k \frac{P(A)}{P(A^c)}} = \frac{k P(A)}{P(A^c) + k P(A)}$$

Now we can view this method as a problem of updating the initial state of belief $P(\cdot)$ using Jeffrey's rule and the soft evidence given by

$$P'(A) = q = \frac{k P(A)}{P(A^c) + k P(A)} \quad \text{where} \quad k = \frac{\text{Odds}'(A)}{\text{Odds}(A)}$$

That is, we have translated a “nothing else considered” specification of soft evidence (a constraint on k) into an “all things considered” specification (a constraint on $P'(A)$).

By [Jeffrey's rule](#) we obtain the new state of belief $P'(\cdot)$ after accommodating soft evidence on event A using a Bayes factor of k :

$$\begin{aligned} P'(B) &= q P(B/A) + (1 - q) P(B/A^c) \\ &= \frac{k P(A)}{P(A^c) + k P(A)} \frac{P(B \cap A)}{P(A)} + \frac{P(A^c)}{P(A^c) + k P(A)} \frac{P(B \cap A^c)}{P(A^c)} \\ &= \frac{k P(B \cap A) + P(B \cap A^c)}{k P(A) + P(A^c)} \end{aligned}$$

Example: this example concerns the alarm of our house and the potential of a burglary. The initial state of belief is given by:

Alarm	Burglary	P()
true	true	0.000095
true	false	0.009999
false	true	0.000005
false	false	0.989901

One day, we receive a call from our neighbour saying that they may have heard the alarm of our house going off. Since our neighbour suffers from a hearing problem, we conclude that our neighbour's testimony increases the odds of the alarm going off by a factor of 4:

$$k = \frac{Odds'(A)}{Odds(A)} = 4$$

where **A**="Alarm=true".

Our goal now is to compute our new belief in a burglary taking place, that is, **P'(B)**, where **B**="Burglary=true".

From the information on this table:

Alarm	Burglary	P()
true	true	0.000095
true	false	0.009999
false	true	0.000005
false	false	0.989901

and $k=4$, by using

$$P'(B) = \frac{k P(B \cap A) + P(B \cap A^c)}{k P(A) + P(A^c)} \quad \text{where} \quad k = \frac{Odds'(A)}{Odds(A)}$$

obtain

$$P'(B) = \frac{4 \times 0.000095 + 0.000005}{4 \times 0.010094 + 0.989906} \approx 0.000374$$

$$\text{where } P(A) = 0.000095 + 0.009999 = 0.010094$$

$$P(A^c) = 1 - P(A) = 1 - 0.010094 = 0.989906$$

(comparing with $P(B)=0.000095+0.000005=0.0001$, **$P'(B)=P(B) \times 3.74$**)

NOTE: The difference between both methods is only in the way the soft evidence about event A is specified:

- By means of $P'(A)$, the final belief assigned to the event, when using the “all things considered” method.
- By means of $k = \text{Odds}'(A) / \text{Odds}(A)$, the relative effect it has on the odds of event A, when using the “nothing else considered” method.

Example: Consider a murder with three suspects: David, Dick and Jane. The state of belief of the investigator

Rich is:

Killer	Rich $P()$
David	2/3
Dick	1/6
Jane	1/6

According to Rich, the odds of David being the killer is 2, since

$$\text{Odds}(\text{David}) = \frac{P(\text{David})}{P(\text{David}^c)} = \frac{2/3}{1 - 2/3} = \frac{2/3}{1/3} = 2$$

Suppose that some new evidence turns up against David. Rich examines the evidence and makes the following statement: “This evidence triples the odds of David being the killer”. Formally, we have **soft evidence** with the following strength ([Bayes factor](#)):

$$k = \frac{Odds'(David)}{Odds(David)} = 3$$

We use the following formula with A=David and k=3:

$$P'(A) = \frac{k P(A)}{P(A^c) + k P(A)} = \frac{3 \times 2/3}{1/3 + 3 \times 2/3} = \frac{6}{7} \approx 0.86 \text{ (86\%)}$$

Rich could have specified the evidence in two ways by saying: “This evidence triples the odds of David being the killer” or “Accepting this evidence leads me to have an 86% belief that David is the killer”.

The first statement could be used with

$$P'(B) = \frac{k P(B \cap A) + P(B \cap A^c)}{k P(A) + P(A^c)} \quad \text{where } A = \text{David} \text{ and } k = 3$$

to compute further beliefs of Rich; for example, his belief in Dick being the killer:

$$\begin{aligned} P'(\text{Dick}) &= \frac{3 \times P(\text{Dick} \cap \text{David}) + P(\text{Dick} \cap \text{David}^c)}{3 \times P(\text{David}) + P(\text{David}^c)} \\ &= \frac{3 \times 0 + P(\text{Dick})}{3 \times P(\text{David}) + P(\text{David}^c)} \\ &= \frac{1/6}{3 \times 2/3 + 1/3} = \frac{1}{14} \end{aligned}$$

or in Jane be the killer, which is also 1/14.

Killer	Rich P'()
David	6/7
Dick	1/14
Jane	1/14

The second statement can be also used for the same purpose but with

$$P'(B) = q P(B/A) + (1 - q) P(B/A^c)$$

to compute further beliefs of Rich, with $A=David$, $q=P'(A)=6/7$. For example, his belief in Dick being the killer:

$$\begin{aligned} P'(Dick) &= \frac{6}{7} P(Dick/David) + \frac{1}{7} P(Dick/David^c) \\ &= \frac{6}{7} \times 0 + \frac{1}{7} \frac{P(Dick \cap David^c)}{P(David^c)} \\ &= \frac{1}{7} \frac{P(Dick)}{P(David^c)} = \frac{1}{7} \times \frac{1/6}{2/6} = \frac{1}{14} \end{aligned}$$

or in Jane be the killer, which is also $1/14$. So, we obtain the same probabilities $P'()$.

However, the difference between the two statements is that the first can be used by some other investigator to update their beliefs based on the new evidence, while the second statement cannot be used as such.

Suppose that Jon is another investigator with the following state of belief, which is different from that held by Rich:

Killer	Jon $P()$
David	1/2
Dick	1/4
Jane	1/4

If Jon were to accept Rich's assessment that the evidence triples the odds of David being the killer, then by using

$$P'(A) = \frac{k P(A)}{P(A^c) + k P(A)}$$

with $A=\text{David}$ and $k=3$ we obtain:

$$P'(\text{David}) = \frac{3 \times P(\text{David})}{P(\text{David}^c) + 3 \times P(\text{David})} = \frac{3 \times 1/2}{1/2 + 3 \times 1/2} = 3/4$$

Analogously, by using

$$P'(B) = \frac{k P(B \cap A) + P(B \cap A^c)}{k P(A) + P(A^c)} \quad \text{where } A = \textit{David} \text{ and } k = 3$$

we can obtain $P'(\textit{Dick})=P'(\textit{Jane})$ in this way:

$$\begin{aligned} P'(\textit{Dick}) &= \frac{3 P(\textit{Dick} \cap \textit{David}) + P(\textit{Dick} \cap \textit{David}^c)}{3 P(\textit{David}) + P(\textit{David}^c)} \\ &= \frac{P(\textit{Dick})}{3 P(\textit{David}) + P(\textit{David}^c)} = \frac{1/4}{3 \times 1/2 + 1/2} = \frac{1}{8} \end{aligned}$$

Killer	Jon $P'()$
David	3/4
Dick	1/8
Jane	1/8

The same evidence that raised Rich's belief from 67% to 86% also raised Jon's belief from 50% to 75%.

The second statement of Rich,

“Accepting this evidence leads me to have about 86% belief that David is the killer”,

is not as meaningful to Jon as it cannot reveal the strength of evidence independently of Rich’s initial beliefs (which we assume are not accessible to Jon).

Hence, **Jon cannot use this statement to update his own beliefs.**

③ Soft evidence as a noisy sensor

One of the most concrete interpretations of soft evidence is in terms of noisy sensors, which is as follows:

Suppose we have some soft evidence that bears on an event A . We can emulate the effect of this soft evidence using a noisy sensor S having two states, with the strength of soft evidence captured by the false positive and false negative rates of the sensor:

- The false positive rate of the sensor, fp , is the belief that the sensor would give a positive reading even though the event A did not occur, that is, $fp = P(S / \text{no } A)$.
- The false negative rate of the sensor, fn , is the belief that the sensor would give a negative reading even though the event A did occur, that is, $fn = P(\text{no } S / A)$.

Suppose now that the sensor reads positive. We want to know the new odds of A given this positive sensor reading. By emulating soft evidence by a positive sensor reading, we have

$$\begin{aligned} Odds'(A) &= \frac{P'(A)}{P'(A^c)} = \frac{P(A/S)}{P(A^c/S)} \\ &= \frac{P(S/A) P(A)}{P(S/A^c) P(A^c)} = \frac{1 - f_n}{f_p} \frac{P(A)}{P(A^c)} \\ &= \frac{1 - f_n}{f_p} Odds(A) \end{aligned}$$

Bayes' theorem

Then, the relative change in the odds of A, the Bayes factor, is indeed a function of only the false positive and the false negative rates of the sensor and is independent of the initial beliefs.

$$\frac{Odds'(A)}{Odds(A)} = \frac{1 - f_n}{f_p}$$

That is, the soft evidence with a Bayes factor of k^+ can be emulated by a positive sensor reading, if the false positive and negative rates of the sensor satisfy

$$k^+ = \frac{1 - f_n}{f_p}$$

Note that the specific false positive and false negative rates are not as important as the above ratio.

Analogously, we would want to know the new odds of A given a negative sensor reading. By emulating soft evidence by a negative sensor reading, we have

$$\begin{aligned} Odds'(A) &= \frac{P'(A)}{P'(A^c)} = \frac{P(A/S^c)}{P(A^c/S^c)} \\ &= \frac{P(S^c/A) P(A)}{P(S^c/A^c) P(A^c)} = \frac{f_n}{1 - f_p} \frac{P(A)}{P(A^c)} \\ &= \frac{f_n}{1 - f_p} Odds(A) \end{aligned}$$

That is, the soft evidence with a Bayes factor of k^- can be emulated by a negative sensor reading, if the false positive and negative rates of the sensor satisfy

$$k^- = \frac{f_n}{1 - f_p}$$

Example: a positive reading from any of the following sensors will have the same impact on beliefs:

- Sensor 1: fp=10% and fn=5%
- Sensor 2: fp=8% and fn=24%
- Sensor 3: fp=5% and fn=52.5%

$$k^+ = 0.95/0.10 = 0.76/0.08 = 0.475/0.05 = \mathbf{9.5}$$

Even though all the sensors have the same k^+ , they have different k^- values:

- Sensor 1: $k^- = 0.05/0.90 \approx \mathbf{0.056}$
- Sensor 2: $k^- = 0.24/0.92 \approx \mathbf{0.261}$
- Sensor 3: $k^- = 0.525/0.95 \approx \mathbf{0.553}$

Note that as long as

$$fp + fn < 1,$$

then $k^+ > 1$ and $k^- < 1$.

This means that a positive sensor reading is guaranteed to increase the odds of the corresponding event and a negative sensor reading is guaranteed to decrease those odds.

Condition $fp + fn < 1$ is satisfied when the false positive and false negative rates are less than 50% each, which is not unreasonable to assume for a sensor model.

The condition, however, can also be satisfied even if one of the rates is $\geq 50\%$.

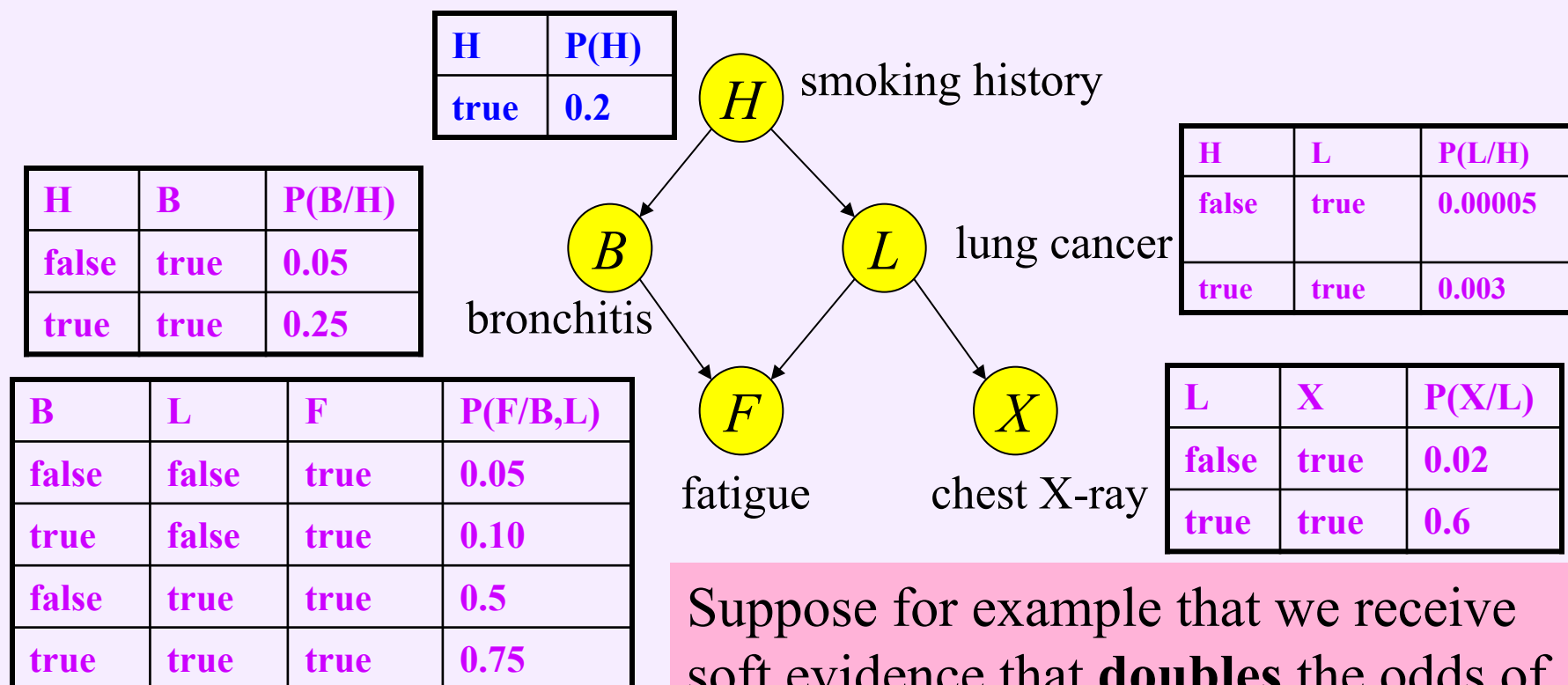
CONCLUSION: Soft evidence on a hypothesis A can be specified using two main methods:

1. The first specifies the final belief in A after accommodating the evidence.
2. The second specifies the relative change in the odds of A due to accommodating the evidence. This relative change in odds is called the **Bayes factor** and can be thought of as providing a strength of evidence that can be interpreted independently of a given state of belief.

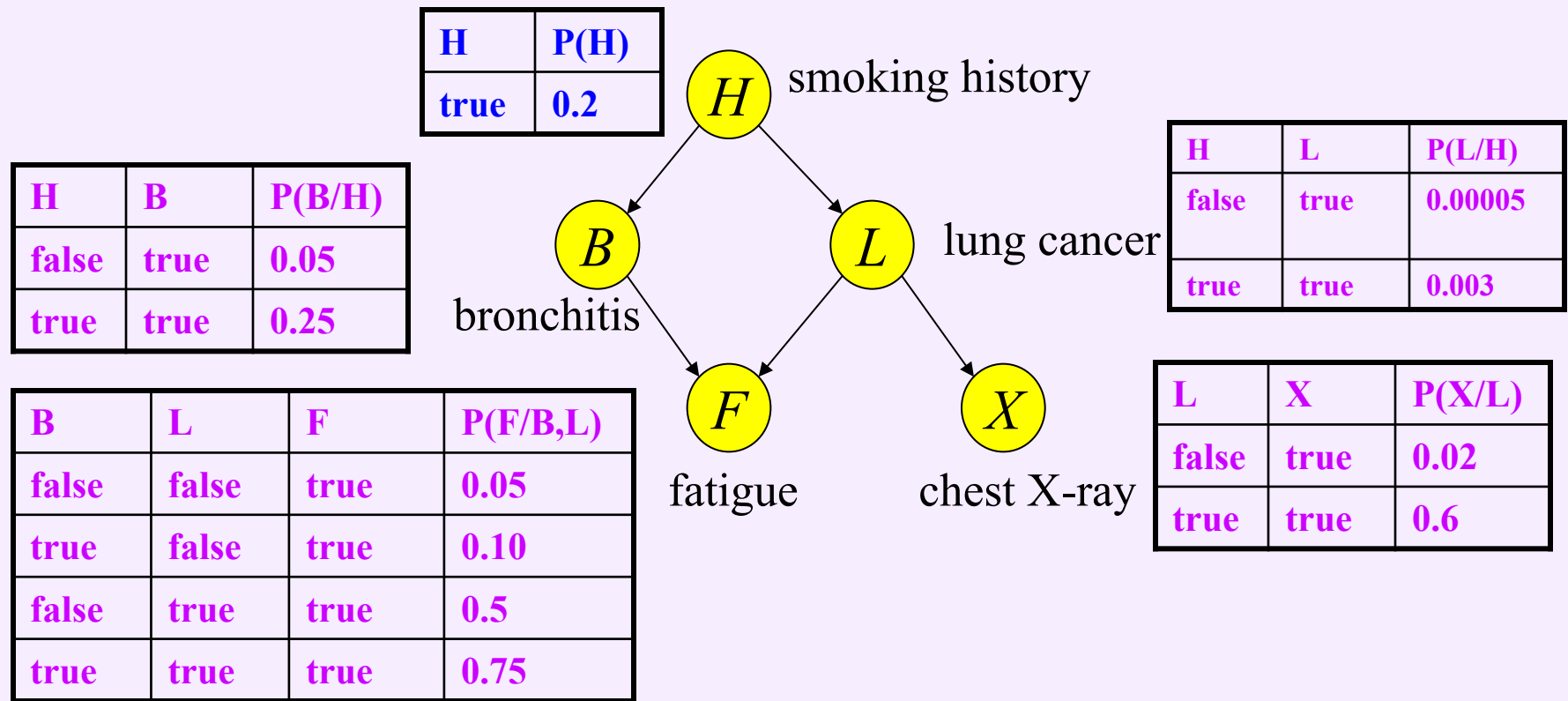
Moreover, the accommodation of soft evidence by a Bayes factor can be emulated by a sensor reading. In particular, for any Bayes factor we can choose the false positive and negative rates of the sensor so its reading will have exactly the same effect on belief as that of the soft evidence. This emulation of soft evidence by hard evidence on an auxiliary variable (**sensor**) is also known as the method of **virtual evidence**.

In the context of Bayesian Networks we can adopt the **virtual evidence** method by adding auxiliary nodes to represent such noisy sensors.

Example 7, Block1 (revisited):



Suppose for example that we receive soft evidence that **doubles** the odds of positive X-ray **or** fatigue.

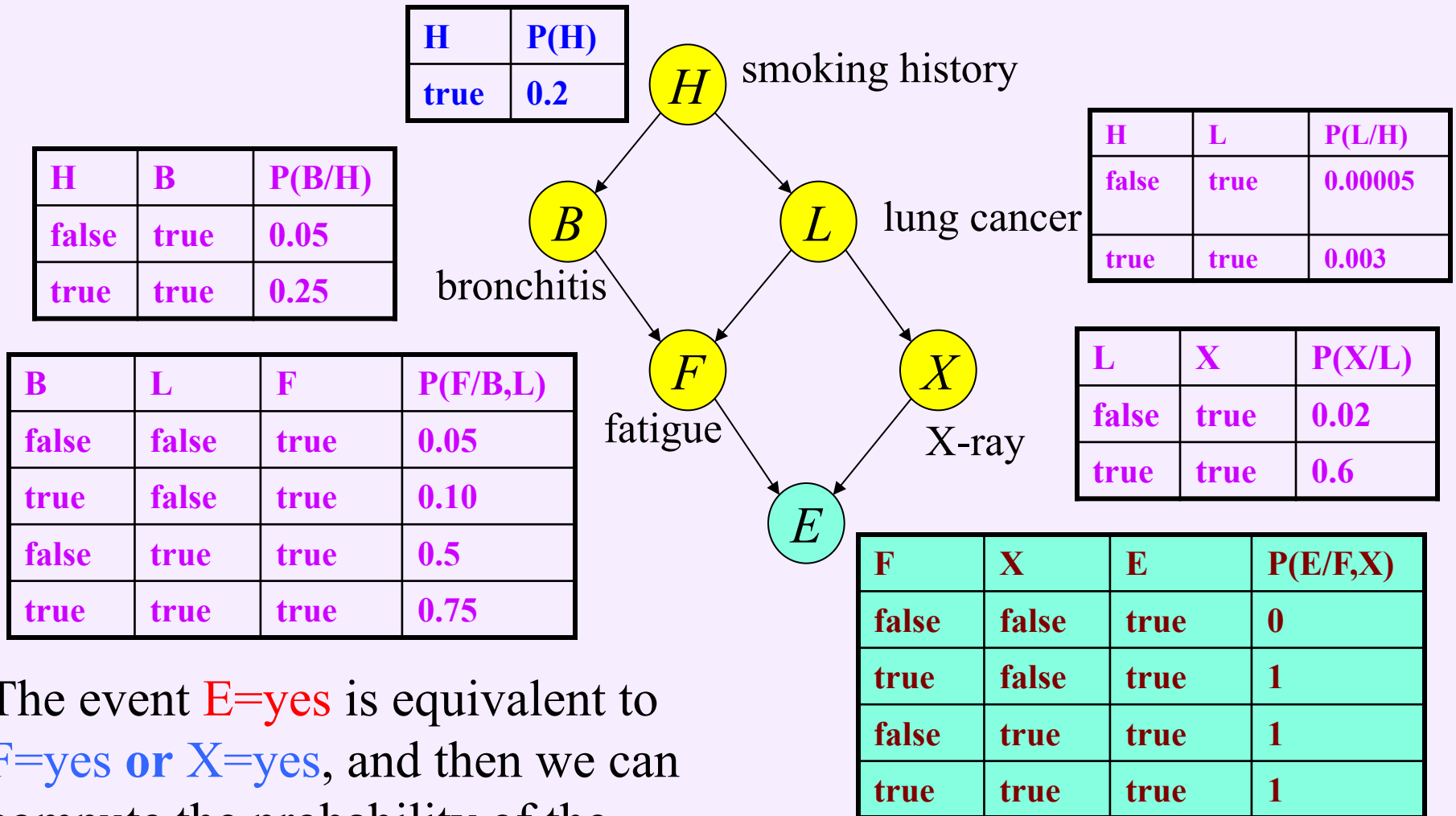


1. Take into account that Bayesian networks tools do not usually provide direct support for computing the probability of arbitrary pieces of evidence as the disjunction

chest X-ray= yes **or** fatigue=yes

But such probabilities can be computed indirectly using the following technique, which consists in using an auxiliary variable E :

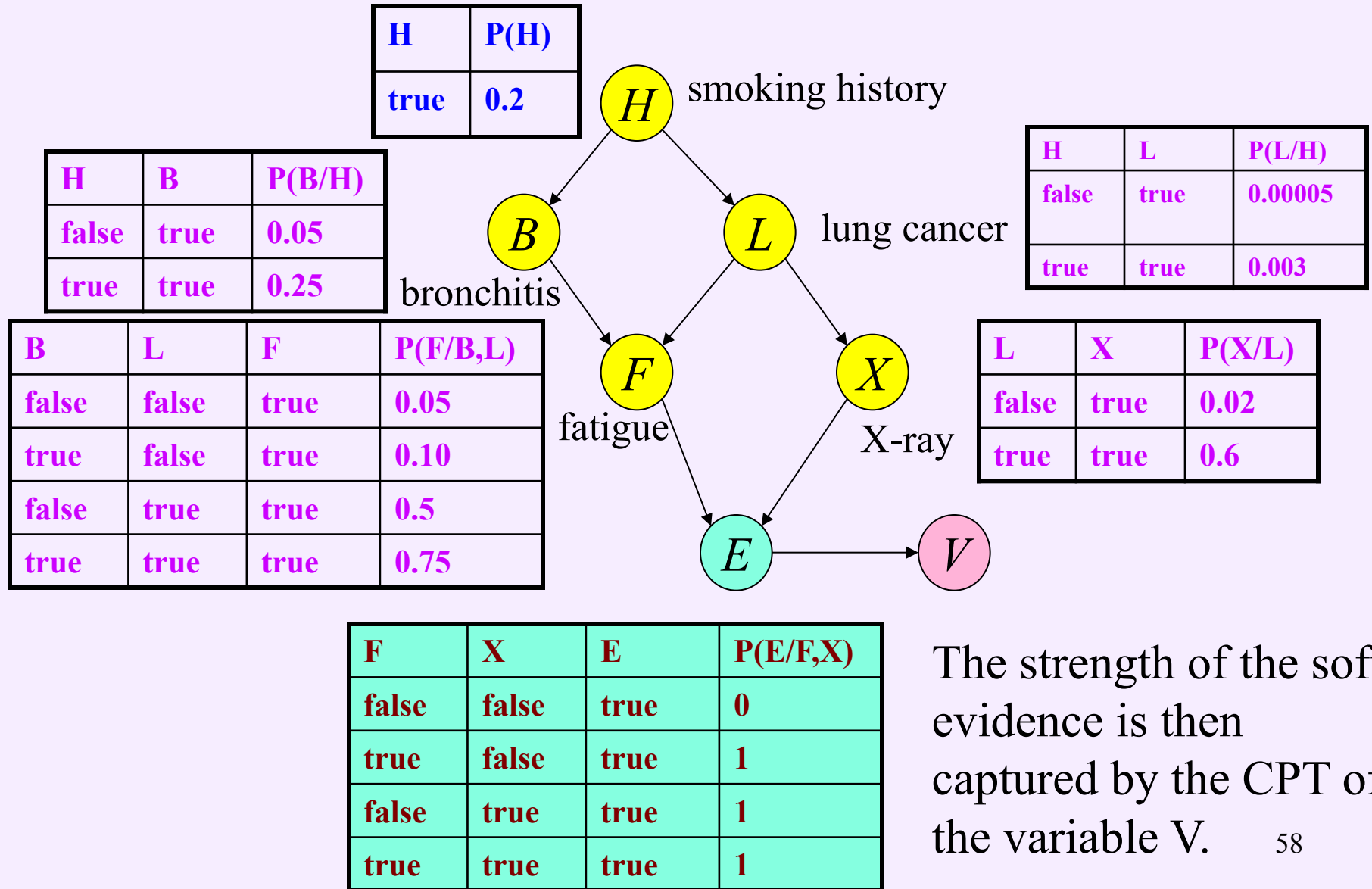
We can add variable E to the network, declare nodes F and X as the parents of E , and then adopt the following CPT for E :



The event $E=\text{yes}$ is equivalent to $F=\text{yes}$ or $X=\text{yes}$, and then we can compute the probability of the latter by computing that of the former.

This is the auxiliary-node method.

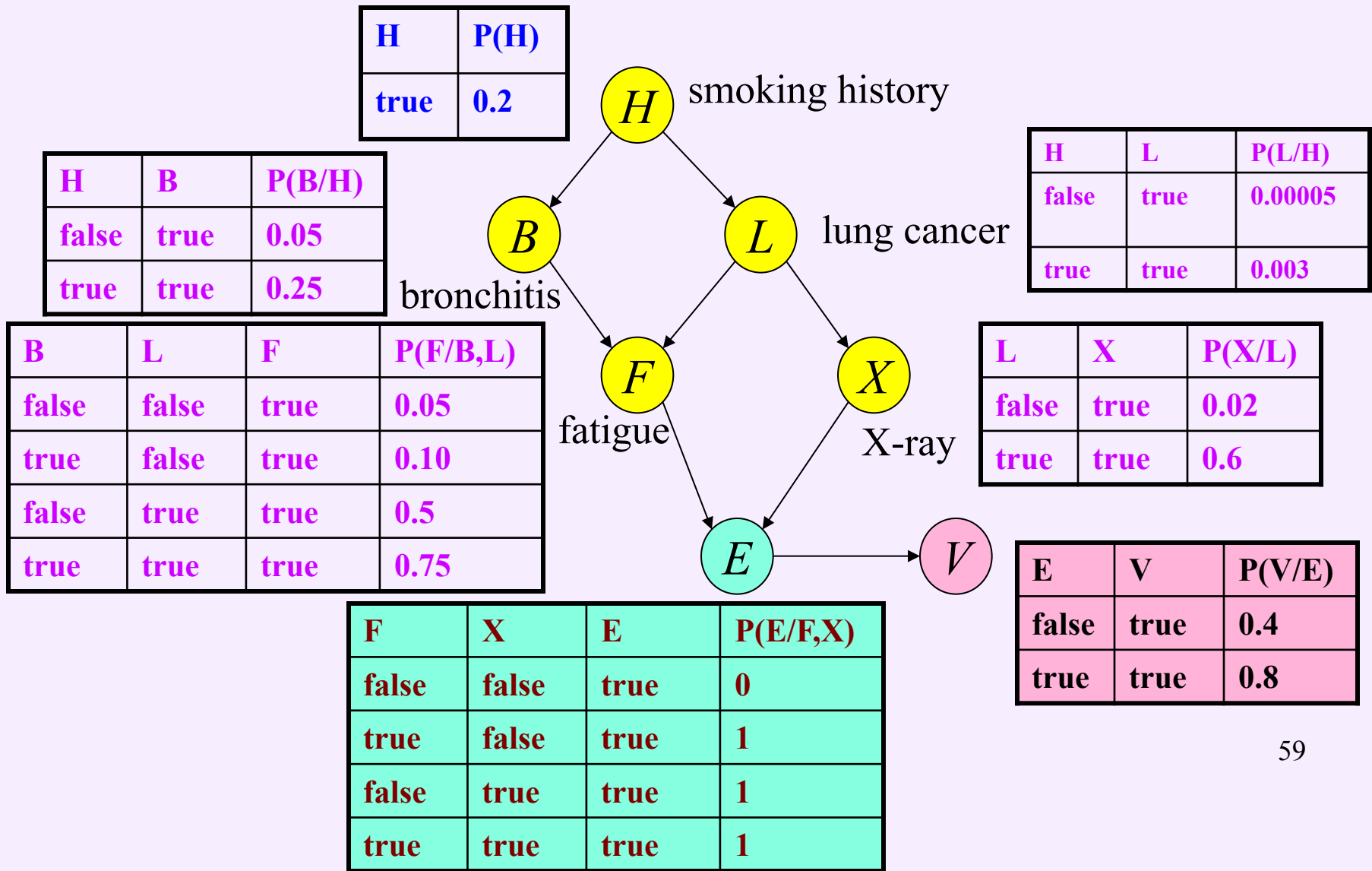
2. We can also represent the soft evidence explicitly by adding another auxiliary variable V to represent the state of a noisy sensor.



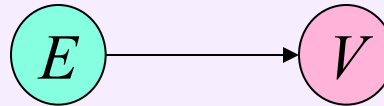
All we have to do is choose a CPT with a **false positive rate fp** and a **false negative rate fn** such that quantifying the strength of the

$$\frac{1 - f_n}{f_p} = k^+$$

where k^+ is the Bayes factor soft evidence, $k^+ = 2$.



F	X	E	P(E/F,X)
false	false	true	0
true	false	true	1
false	true	true	1
true	true	true	1



E	V	P(V/E)
false	true	0.4
true	true	0.8

Justification:

$$\begin{aligned}
 1 - f_n &= 1 - P(V = \text{false} / E = \text{true}) = P(V = \text{true} / E = \text{true}) \\
 f_p &= P(V = \text{true} / E = \text{false}) \\
 \frac{1 - f_n}{f_p} &= \frac{P(V = \text{true} / E = \text{true})}{P(V = \text{true} / E = \text{false})} = 2
 \end{aligned}$$

So we can choose any numbers in the CPT of V if

$$P(V=\text{true} / E=\text{true}) = 2 P(V=\text{true} / E=\text{false})$$

We can then accommodate the soft evidence by setting the value of auxiliary variable V to **TRUE**:

With gRain:

```
# All nodes are boolean variables:
tf<-c("true","false")
# Specify the CPTs:
node.H<-cptable(~ H, values=c(2,8),levels=tf)
node.B<-cptable(~ B + H, values=c(25,75,5,95), levels=tf)
node.L<-cptable(~ L + H, values=c(3,997,5,99995), levels=tf)
node.F<-cptable(~ F + L + B, values=c(75,25,1,9,5,5,5,95), levels=tf)
node.X<-cptable(~ X + L, values=c(6,4,2,98), levels=tf)
node.E<-cptable(~ E + X + F, values=c(1,0,1,0,1,0,0,1), levels=tf)
node.V<-cptable(~ V + E, values=c(8,2,4,6), levels=tf)
# Create an intermediate representation of the CPTs:
Plist<-compileCPT(list(node.H,node.B,node.L,node.F,node.X,node.E,node.V))
# Create network:
netgrain<-grain(plist)
summary(netgrain)
plot(netgrain)
```

With **gRain**:

```
# The marginal probability (EXACT!!!):
```

```
#
```

```
querygrain(netgrain,nodes=c("E"), type="marginal")
```

The prior marginal of variable E is:

E	P(E)
false	0.92611441
true	0.07388559

```
netgrain.2<-setEvidence(netgrain,nodes=c("V"), states=c("true"))
```

```
# New marginal:
```

```
querygrain(netgrain.2,nodes=c("E"), type="marginal")
```

Then the posterior marginal of variable E given that V=true is:

E	P'(E)=P(E/V=true)
false	0.86239578
true	0.13760422

The ratio of odds is then:

$$\frac{\text{Odds}(E = \text{true}/V = \text{true})}{\text{Odds}(E = \text{true})} = \frac{0.13760422/0.86239578}{0.07388559/0.92611441} \approx 2$$

Hence, the hard evidence **V=true** leads to doubling the odds of **E=true**, as expected!